

Using Max-Algebra Linear Models in the Representation of Queueing Systems*

Nikolai K. Krivulin[†]

Abstract

The application of the max-algebra to describe queueing systems by both linear scalar and vector equations is discussed. It is shown that these equations may be handled using ordinary algebraic manipulations. Examples of solving the equations representing the $G/G/1$ queue and queues in tandem are also presented.

1 Introduction

Max-algebra $[1, 2]$ is the system $(\mathbb{R} \cup \{\varepsilon\}, \oplus, \otimes)$, where

$$\varepsilon = -\infty, \quad x \oplus y = \max(x, y), \quad x \otimes y = x + y \quad \forall x, y \in \mathbb{R}.$$

It has the following properties which can be easily verified

$$\begin{aligned} \forall x, y, z \in \mathbb{R} \quad & x \oplus (y \oplus z) = (x \oplus y) \oplus z, \quad x \oplus y = y \oplus x, \\ & x \otimes (y \otimes z) = (x \otimes y) \otimes z, \quad x \otimes y = y \otimes x, \\ & x \otimes (y \oplus z) = (x \otimes y) \oplus (x \otimes z), \\ & x \oplus \varepsilon = x, \quad x \oplus x = x, \\ & x \otimes e = x, \quad \text{where } e = 0. \end{aligned}$$

In the max-algebra these properties allow ordinary algebraic manipulation of linear expressions to be performed under the usual conventions regarding brackets and precedence of \otimes over \oplus . Moreover, the scalar max-algebra is extended to the max-algebra of vectors in the regular way. To emphasize parallels between conventional linear algebra and the max-algebra, similar notations are used for the iterated operations \oplus and \otimes

$$\sum_{\oplus, i=1}^n x_i = x_1 \oplus \cdots \oplus x_n, \quad \prod_{\otimes, i=1}^n x_i = x_1 \otimes \cdots \otimes x_n.$$

*Proc. 5th SIAM Conf. on Applied Linear Algebra, Snowbird, UT, June 15-18, 1994, 155–160

[†]Postdoctoral Research Fellow, Faculty of Mathematics and Mechanics, St.Petersburg State University, Bibliotechnaya sq.2, Petrodvorets, St.Petersburg, 198904 Russia

The max-algebra theory is currently under investigation. There are a number of classical algebraic results reformulated and proved in this algebra. Specifically, the eigenvalue problem has been solved, and analogues of Cramer's rule and the Cayley-Hamilton theorem have been found (see survey papers [1, 2]). Moreover, as a research tool in studying practical problems, the max-algebra finds expanding applications in many fields of operations research and optimization, including the analysis and performance evaluation of discrete event dynamic systems [1, 3, 2].

Although max-algebra models were successfully applied to investigate certain classes of discrete event dynamic systems, the models of queues have received little or no attention. The purpose of this paper is to show how queueing systems may be described using the max-algebra approach by linear algebraic equations. As illustrations of handling these algebraic models, the solutions of the equations representing the $G/G/1$ queue and queues in tandem are presented.

2 A Linear Algebraic Model for the $G/G/1$ Queue

We start with the linear max-algebra representation of the $G/G/1$ queue which provides the basis for more complicated models of queueing systems. In the analysis of queueing systems, it is common to apply recursive equations to describe their dynamics analytically. Such equations are normally written in terms of recursions for the arrival and departure times of customers, and involve the operations of maximum and addition [3, 4, 5].

To set up the equations that represent the $G/G/1$ queue in the ordinary way, consider a single server queue with infinite buffer capacity. Once a customer arrives into the system, he occupies the server provided that it is free. If the customer finds the server busy, he is placed into the buffer and has to wait until the service of all his predecessors completes.

Denote the interarrival time between the k th customer and his predecessor by α_k , and the service time of the k th customer by τ_k . Furthermore, let $A(k)$ and $D(k)$ be the arrival and departure times of the k th customer, respectively. As is customary, we assume that $\alpha_k \geq 0$ and $\tau_k \geq 0$ are given parameters, whereas $A(k)$ and $D(k)$ are unknown variables. With the conditions that the queue starts operating at time zero and it is free at the initial time, one can readily represent the system dynamics by the set of equations [3, 4, 5]

$$\begin{aligned} A(k) &= A(k-1) + \alpha_k, \\ D(k) &= \max(A(k), D(k-1)) + \tau_k. \end{aligned}$$

Let us now replace the usual operation symbols by those of the max-

algebra and rewrite the equations in their equivalent form as

$$A(k) = \alpha_k \otimes A(k-1), \quad (1)$$

$$D(k) = \tau_k \otimes (A(k) \oplus D(k-1)). \quad (2)$$

Under the properties of the operation \oplus and \otimes these equations could be handled much as if they were ordinary linear equations in the conventional algebra. Specifically, applying a usual technique to solve the equations for the unknown variables $A(k)$ and $D(k)$, we get

$$A(k) = \alpha_1 \otimes \cdots \otimes \alpha_k, \quad D(k) = \sum_{\oplus, i=1}^k \alpha_1 \otimes \cdots \otimes \alpha_i \otimes \tau_i \otimes \cdots \otimes \tau_k. \quad (3)$$

2.1 The Matrix Representation

To produce a matrix representation for the $G/G/1$ queue let us first define the vector $\mathbf{D}(k) = (D_0(k), D_1(k))^T$ with components $D_0(k) = A(k)$, $D_1(k) = D(k)$, and replace the symbols α_k and τ_k by τ_{0k} and τ_{1k} , respectively, $k = 1, 2, \dots$. It is convenient to preassign $D_0(0) = D_1(0) = e$, and $D_0(k) = D_1(k) = \varepsilon$ for all $k < 0$. With the new notations, the equations (1-2) may be rewritten as

$$D_0(k) = \tau_{0k} \otimes D_0(k-1), \quad (4)$$

$$D_1(k) = \tau_{1k} \otimes (D_0(k) \oplus D_1(k-1)). \quad (5)$$

Substitution of (4) into (5) and the implementation of distributivity of \otimes over \oplus give

$$D_0(k) = \tau_{0k} \otimes D_0(k-1),$$

$$D_1(k) = \tau_{1k} \otimes \tau_{0k} \otimes D_0(k-1) \oplus \tau_{1k} \otimes D_1(k-1).$$

We may now represent the model in matrix notations by the equation

$$\mathbf{D}(k) = T_k \otimes \mathbf{D}(k-1), \quad (6)$$

where the transition matrix is defined as

$$T_k = \begin{pmatrix} \tau_{0k} & \varepsilon \\ \tau_{1k} \otimes \tau_{0k} & \tau_{1k} \end{pmatrix}.$$

3 Linear Models of $G/G/1$ Queues in Tandem

In this section we extend the max-algebra linear models to cover systems of $G/G/1$ queues operating in tandem. As the basic system of this type, we first consider a series of n queues with infinite buffers. Each customer that

arrives into this system is initially placed in the buffer at the 1st server and then has to pass through all the queues consecutively. Upon the completion of his service at server i , the customer is instantaneously transferred to queue $i + 1$, $i = 1, \dots, n - 1$. The customer leaves the system after his service completion at the n th server.

For the tandem queueing system the equations (4-5) can be easily generalized as

$$D_0(k) = \tau_{0k} \otimes D_0(k-1), \quad (7)$$

$$D_i(k) = \tau_{ik} \otimes (D_{i-1}(k) \oplus D_i(k-1)), \quad i = 1, \dots, n. \quad (8)$$

where $D_i(k)$ and τ_{ik} denote the departure time and the service time of k th customer at server i , respectively.

Let $\mathbf{D}(k) = (D_0(k), \dots, D_n(k))^T$ be the vector of the k th departure times in the system. Similarly as in the case of the $G/G/1$ queue, we may write the vector equation representing the tandem queueing system in the form (6) with the lower triangular transition matrix

$$T_k = \begin{pmatrix} & \tau_{0k} & & \varepsilon & & \varepsilon & \dots & \varepsilon \\ & \tau_{1k} \otimes \tau_{0k} & & \tau_{1k} & & \varepsilon & \dots & \varepsilon \\ & \vdots & & \vdots & & & \ddots & \vdots \\ \tau_{n-1k} \otimes \dots \otimes \tau_{0k} & \tau_{n-1k} \otimes \dots \otimes \tau_{1k} & \tau_{n-1k} \otimes \dots \otimes \tau_{2k} & & & & & \varepsilon \\ \tau_{nk} \otimes \dots \otimes \tau_{0k} & \tau_{nk} \otimes \dots \otimes \tau_{1k} & \tau_{nk} \otimes \dots \otimes \tau_{2k} & \dots & \tau_{nk} & & & \end{pmatrix}.$$

Furthermore, we may find the solution of the set of recursive equations (7-8) as an extension of (3). With usual algebraic manipulations, it can be arrived at ([4])

$$D_n(k) = \sum_{\oplus, 1 \leq i_1 \leq \dots \leq i_n \leq n} \left(\prod_{\otimes, j=1}^{i_1} \tau_{0j} \otimes \prod_{\otimes, j=i_1}^{i_2} \tau_{1j} \otimes \dots \otimes \prod_{\otimes, j=i_n}^k \tau_{nj} \right), \quad k = 1, 2, \dots$$

3.1 Tandem Queues with Finite Buffers

Suppose now that the buffers of servers in the tandem system described above have finite capacity. The feature of queueing systems with limited buffers is that their servers may be blocked according to one of the blocking rules [3]. In this paper we restrict our consideration to *manufacturing* blocking which is most commonly encountered in practice. Under this type of blocking, if the i th server upon completion of a service sees the buffer of the $(i + 1)$ st server full, it cannot be unoccupied and has to be busy until the $(i + 1)$ st server completes its current service to provide a free space in its buffer.

Consider a queueing system with n servers in tandem, and assume the buffer at the i th server, $i = 2, \dots, n$, to be of the capacity b_i , $0 \leq b_i < \infty$.

We suppose that the buffer of the 1st server, as the input buffer of the system, is infinite. Since the customers leave the system upon their service completion at the n th server, this server cannot be blocked.

It is not difficult to understand that the k th completion time at the i th server, $i = 1, \dots, n-1$, can be represented in usual form by the recursive equation [3, 4]

$$D_i(k) = \max(\max(D_{i-1}(k), D_i(k-1)) + \tau_{ik}, D_{i+1}(k - b_{i+1} - 1)).$$

Using max-algebra notations, the complete set of linear equations describing the finite buffers tandem queueing system with manufacturing blocking is written as

$$D_0(k) = \tau_{0k} \otimes D_0(k-1), \quad (9)$$

$$D_i(k) = \tau_{ik} \otimes (D_{i-1}(k) \oplus D_i(k-1)) \oplus D_{i+1}(k - b_{i+1} - 1), \\ i = 1, \dots, n-1, \quad (10)$$

$$D_n(k) = \tau_{nk} \otimes (D_{n-1}(k) \oplus D_n(k-1)). \quad (11)$$

Although it is evident that we are dealing here with a linear model once again, handling the model in its general form (9-11) requires rather cumbersome algebraic manipulations. Therefore, let us consider more thoroughly a simple example of a system with $n = 2$, $b_2 = 0$. The equations (9-11) are reduced to

$$D_0(k) = \tau_{0k} \otimes D_0(k-1), \quad (12)$$

$$D_1(k) = \tau_{1k} \otimes (D_0(k) \oplus D_1(k-1)) \oplus D_2(k-1), \quad (13)$$

$$D_2(k) = \tau_{2k} \otimes (D_1(k) \oplus D_2(k-1)). \quad (14)$$

Going to matrix notations, we arrive at the linear equation

$$\mathbf{D}(k) = \tilde{T}_k \otimes \mathbf{D}(k-1),$$

with

$$\tilde{T}_k = \begin{pmatrix} & \tau_{0k} & \varepsilon & \varepsilon \\ & \tau_{1k} \otimes \tau_{0k} & \tau_{1k} & e \\ \tau_{2k} \otimes \tau_{1k} \otimes \tau_{0k} & \tau_{2k} \otimes \tau_{1k} & \tau_{2k} & \end{pmatrix}.$$

The above representation of the transition matrix for the system with two servers is easily extended to the case of the system with n servers and $b_i = 0$, $i = 2, \dots, n$

$$\tilde{T}_k = \begin{pmatrix} & \tau_{0k} & \varepsilon & \varepsilon & \dots & \varepsilon \\ & \tau_{1k} \otimes \tau_{0k} & \tau_{1k} & e & & \varepsilon \\ & \vdots & \vdots & & \ddots & \\ \tau_{n-1k} \otimes \dots \otimes \tau_{0k} & \tau_{n-1k} \otimes \dots \otimes \tau_{1k} & \tau_{n-1k} \otimes \dots \otimes \tau_{2k} & & & e \\ \tau_{nk} \otimes \dots \otimes \tau_{0k} & \tau_{nk} \otimes \dots \otimes \tau_{1k} & \tau_{nk} \otimes \dots \otimes \tau_{2k} & \dots & \tau_{nk} & \end{pmatrix}.$$

Note that the matrices \tilde{T}_k and T_k differ only in elements of the upper diagonal adjacent to the main diagonal. In \tilde{T}_k these elements become equal to e , excluding the one of row 0 which remains equaled ε .

Now we return to the example so as to present the solution of the recursive equations (12-14). After usual algebraic manipulations one can obtain

$$\begin{aligned} D_1(k) &= \sum_{i=1}^k \oplus \left(\prod_{j=1}^i \tau_{0j} \otimes \tau_{1i} \otimes \prod_{j=i+1}^k (\tau_{1j} \oplus \tau_{2j-1}) \right), \\ D_2(k) &= \tau_{2k} \otimes D_1(k). \end{aligned}$$

3.2 Closed Systems of $G/G/1$ Queues

Consider a closed tandem system of n queues with infinite buffers. We assume that the customers after their service completion at the n th server return to the 1st server for a new cycle of service. There are the finite number of customers circulating through the system, at the initial time all the customers are placed in the buffer of the 1st server.

Let us denote the number of customers in the system by c . With the condition $D_n(k) = \varepsilon$ for all $k < 0$, we may modify (7-8) to write the set of equations for the closed system without specifying $D_0(k)$, in the form

$$\begin{aligned} D_1(k) &= \tau_{1k} \otimes (D_n(k - c) \oplus D_1(k - 1)), \\ D_i(k) &= \tau_{ik} \otimes (D_{i-1}(k) \oplus D_i(k - 1)), \quad i = 2, \dots, n. \end{aligned}$$

To produce a matrix representation, we now define the vector of departure times as $\mathbf{D}(k) = (D_1(k), \dots, D_n(k))^T$. The vector equation associated with the closed tandem queueing system with c customers is represented as

$$\mathbf{D}(k) = R_k \otimes \mathbf{D}(k - 1) \oplus S_k \otimes \mathbf{D}(k - c),$$

where

$$\begin{aligned} R_k &= \begin{pmatrix} \tau_{1k} & \varepsilon & \dots & \varepsilon \\ \tau_{2k} \otimes \tau_{1k} & \tau_{2k} & & \varepsilon \\ \vdots & \vdots & \ddots & \\ \tau_{nk} \otimes \dots \otimes \tau_{1k} & \tau_{nk} \otimes \dots \otimes \tau_{2k} & \dots & \tau_{nk} \end{pmatrix}, \\ S_k &= \begin{pmatrix} \varepsilon & \dots & \varepsilon & \tau_{1k} \\ \varepsilon & \dots & \varepsilon & \tau_{2k} \otimes \tau_{1k} \\ \vdots & & \vdots & \vdots \\ \varepsilon & \dots & \varepsilon & \tau_{nk} \otimes \dots \otimes \tau_{1k} \end{pmatrix}. \end{aligned}$$

In conclusion, consider an example of a closed system with $n = 2$, $c = 2$. We have the following linear equations representing this system

$$\begin{aligned} D_1(k) &= \tau_{1k} \otimes (D_1(k - 1) \oplus D_2(k - 2)), \\ D_2(k) &= \tau_{2k} \otimes (D_1(k) \oplus D_2(k - 1)). \end{aligned}$$

Traditional methods of solving linear recursions give the solution

$$D_1(k) = \tau_{11} \otimes \prod_{\otimes, i=1}^{k-2} (\tau_{1i+1} \oplus \tau_{2i}) \otimes \tau_{1k},$$

$$D_2(k) = \tau_{11} \otimes \prod_{\otimes, i=1}^{k-1} (\tau_{1i+1} \oplus \tau_{2i}) \otimes \tau_{2k}.$$

References

- [1] R. A. Cuninghame-Green, “Minimax algebra and applications,” *Fuzzy Sets and Syst.* **41** no. 3, (1991) 251–267.
- [2] G. J. Olsder, “About difference equations, algebras and discrete events,” vol. 31 of *CWI syllabus*, pp. 180–209. CWI, Amsterdam, 1992.
- [3] A. G. Greenberg, B. D. Lubachevsky, and I. Mitran, “Algorithms for unboundedly parallel simulations,” *ACM Trans. Comput. Syst.* **9** no. 3, (Aug., 1991) 201–221.
- [4] N. K. Krivulin, *Optimization of Discrete Event Dynamic Systems by Using Simulation, PhD Dissertation*. St. Petersburg University, St. Petersburg, 1990. (in Russian).
- [5] N. Krivulin, “Unbiased estimates for gradients of stochastic network performance measures,” *Acta Applic. Math.* **33** (1993) 21–43, [arXiv:1210.5418 \[math.OC\]](#).